

Network Extender Module

This application claims priority under 35 USC §(e)(1) of British Application Number 0021873.5 filed September 6, 2000.

5

The present invention relates to communications over a data network and more especially to voice telephony over a conventional Ethernet network.

10 Voice Over Internet Protocol (VOIP) is a protocol that facilitates the use of a data network for voice telephony. A VOIP telephone system can be incorporated into a computer workstation but most users prefer a separate handset that is more in keeping with a conventional telephone system. In  
15 order to use a separate handset, the VOIP telephone and the computer workstation can share a network connection, or else they will require separate network connections. Many computer workstations are equipped with only one network connection and so it is advantageous to provide a system in  
20 which the computer workstation and the VOIP telephone can share the same network connection. Circuit modules that enable more than one system to connect to a network using the same network connection are known as network extender modules.

25

A known half-duplex VOIP system uses the configuration shown in Figure 1. A VOIP telephone 1, computer workstation 2 and network 3 are each connected to a repeater 4. When one of the VOIP telephone 1, computer workstation 2 or network 3  
30 wishes to transmit data, that device must acquire control of the repeater 4. Data can then be transferred from that device to the repeater 4. The repeater 4 then repeats that incoming data to the other two ports. In this way, the repeater 4 allows data to be transferred from the computer

workstation 2 to the network 3, from the network to the computer workstation, from the VOIP telephone 1 to the network and from the network to the VOIP telephone. Further, the repeater 4 allows control data to pass between the computer workstation 2 and the VOIP telephone 1.

A major disadvantage with the configuration of Figure 1 is that the connection is half-duplex (i.e. in only one direction at a time). Full-duplex connections on the other hand have twice the bandwidth of half-duplex connections. Further, for two-way traffic, there is no need for the two ports to compete for control of the network when a full-duplex connection is used; both ports can transmit at the same time.

It is an object of the present invention to provide an improved means of adding, for example, a VOIP telephone connection to a conventional computer workstation to data network connection using a single network connection.

An embodiment of the network extender module will now be described by way of example only with reference to the accompanying drawings, of which:

FIGURE 1 is a block diagram of a known half-duplex VOIP telephone system;

FIGURE 2 is a block diagram of a full-duplex VOIP telephone system using a three-port switch;

FIGURE 3 is a graphical representation of the data flow in the telephone system of Figure 2;

FIGURE 4 is a block diagram of a full-duplex VOIP telephone system including a network extender module in accordance with the present invention;

FIGURE 5 is a block diagram of the VOIP telephone system of Figure 4 implemented using RAM.

Figure 2 shows a configuration that provides full-duplex connections. In Figure 2, the repeater 4 of Figure 1 is replaced by a three-port switch 5. Each port of the switch provides two connections - a line in and a line out - in order to provide a full duplex connection. A switch allows data received at a port to be routed in general to any other one of its ports; the internal structure of the switch needed for that routing is complex and will now be described.

Two known methods of handling the transfer of broadcast packets of data in the switch 5 are (1) making a copy of each incoming broadcast packet for each destination port and (2) using data structures to manage packet data such that a single copy of each broadcast packet can be referred to by each destination port. The first method requires relatively high bandwidth since multiple copies of the same data will be created. The second method requires memory management to ensure that the single copy of the data only exists until all destination ports have sent that data. One possible implementation is to use linked lists. The linked lists form queues of packets, for example broadcast packets, for each port of the switch, with packets able to exist on more than one queue. Each packet can only be deleted once it has been un-linked for all of its destination queues. This last option is more complicated.

The data traffic in a VOIP switch module, such as that of Figure 2, consists of three channels:

1. Computer data traffic between the network 3 and the computer workstation 2;

2. VOIP traffic between the VOIP telephone 1 and the network 3;
3. Occasional control traffic between the VOIP telephone 1 and the computer workstation 2.

5

The three data traffic channels are shown diagrammatically in Figure 3.

The traffic between the network 3 and the computer workstation 2 comprises the bulk of the traffic that passes through the switch 5 with the control traffic between the VOIP telephone 1 and the computer workstation 2 being relatively infrequent. The present invention makes use of the restricted data flow, in particular by simplifying the overall buffering requirements.

An embodiment of a network extender module according to the present invention is shown in Figure 4. In use, the network extender module 6 of Figure 4 replaces the switch 5 of Figure 2. The network extender module 6 comprises two media access controllers (MACs) 7 and 8, two buffer managers 9 and 10 and six first-in-first-out (FIFO) buffers 11 to 16.

MAC 7 controls the flow of data to and from the network 3, MAC 8 controls the flow of data to and from the computer workstation 2. Buffer managers 9 and 10 control data flow to and from the VOIP telephone 1 but buffer manager 9 handles the traffic between the network 3 and the VOIP telephone 1 and buffer manager 10 handles the traffic between the computer workstation 2 and the VOIP telephone. FIFOs 11 to 16 are connected between MACs 7 and 8 and buffer managers 9 and 10 in order to regulate the flow of data between them. FIFO 11 regulates the flow of data from MAC 7 to MAC 8, FIFO 12 regulates the flow of data from MAC 8 to MAC 7, FIFOs 13

and 14 regulate the flow of data from MAC 7 to buffer manager 9 and from buffer manager 9 to MAC 7 respectively and FIFOs 15 and 16 regulate the flow of data from MAC 8 to buffer manager 10 and from buffer manager 10 to MAC 8 respectively.

5

Each of MACs 7 and 8 perform the data serialisation and de-serialisation and handle the network access protocols that are required for transferring data to and from the network 3 and to and from the computer workstation 2, using preferably standard network protocols. Buffer managers 9 and 10 transfer packets of data to and from the VOIP telephone 1 and MACs 7 and 8. Data transferred between the VOIP telephone 1 and the network extender module 6 is generally via a physically short connection and is of low bandwidth.

15 Therefore, it is not necessary there to use data transmission techniques. The buffer managers 9 and 10 are not required to handle network access protocols or to perform data serialisation and de-serialisation; the data can be passed to and from the VOIP telephone in parallel and since the VOIP

20 telephone may be designed specially to work with the network extender module and indeed may be part of the same unit a non-standard transmission protocol may be used. The buffer managers 9 and 10 are therefore in the preferred embodiment much simpler devices than MACs 7 and 8.

25

MAC 7 takes data from FIFOs 12 and 14 (one at a time) and MAC 8 takes data from FIFOs 11 and 16. The data is then buffered and passed to the network (in the case of MAC 7) or the computer workstation (in the case of MAC 8). MACs 7 and 8

30 also receive data from the network 3 and the computer workstation 2 respectively and route that data to one or both of FIFOs 11 and 13 (in the case of MAC 7) and FIFOs 12 and 15 (in the case of MAC 8). Incoming data that is addressed to a particular source (unicast data) is routed to the appropriate

FIFO. Data that is not so addressed (broadcast data) is routed to both of the relevant FIFOs. Control circuitry (not shown) responsive to relevant address portions of the incoming packets activates the MACs and FIFOs appropriately to achieve this. This circuitry is preferably connected to be responsive to the address portions when the packet is in the MAC into which it is received from the network or the computer workstation. Preferably, the address recognition takes place either prior to or during the transfer of data from the MACs to the FIFOs. If the address recognition occurs during the transfer of data, the FIFOs pointers may be manipulated in order to purge written data that is unwanted.

Buffer managers 9 and 10 take data from FIFOs 13 and 15 respectively, buffer the data and transfer it to the VOIP telephone 1. Buffer managers 9 and 10 also take data from the VOIP telephone 1, buffer the data and pass it to FIFOs 14 and 16 respectively. The VOIP telephone 1 is preferably constructed to send and receive voice and control data for the network to and from buffer manager 9 and to send and receive control data to and from the computer workstation via buffer manager 10. The connection between VOIP telephone 1 and network extender module 6 may be a single bus on which data to and from the computer workstation 2 and data to and from the network 3 would be interleaved. The interleaved data is then routed as required by a single buffer manager, that buffer manager performing the functions of buffer managers 9 and 10 by sequentially servicing both sets of FIFOs 13 and 14, and, 15 and 16. Otherwise the circuitry of VOIP telephone is conventional.

Data entering MAC 7 and intended for the network may arrive from either the VOIP telephone 1 via FIFO 14, or from the computer workstation 2 via FIFO 12. MAC 7 therefore requires

a method for selecting the next data, especially when both FIFO 12 and FIFO 14 are ready to transmit a packet of data. The simplest form of control gives either the VOIP telephone 1 or the computer workstation 2 priority. Under normal  
5 circumstances, the VOIP telephone 1 should have priority since the user will readily notice an increase in the latency of telephonic data. If the VOIP telephone 1 is given priority at all times, as is preferable, when data from FIFO 12 and FIFO 14 are ready to be sent to MAC 7 at the same  
10 time, the MAC simply transmits the data packet from FIFO 14 and signals FIFO 12 to prevent it from sending its packet. This data transfer will continue until FIFO 14 is empty; MAC 7 would then take data from FIFO 12. The same method is used by MAC 8 to handle data arriving simultaneously from FIFOs 11  
15 and 16, preferably with the VOIP telephone 1 having priority.

It will be appreciated that other packet priority schemes could be devised and the invention is not limited to the examples disclosed herein.

20

Such possible schemes include round robin prioritisation, where the MACs alternate between which of two competing sets of data they transmit and bandwidth allocation, where prioritisation is based on the amount of bandwidth consumed  
25 by the transmissions. In the present case, however, giving the VOIP telephone 1 priority at all times is preferable for at least three reasons. First, it meets the low latency requirements of the VOIP telephone data. Second, it will not greatly hinder the transfer of data between the network 3 and  
30 the computer workstation 2, due to the low bandwidth of the VOIP telephone data. Third, it is a very straightforward scheme to implement.

In addition to receiving data from two FIFOs, MACs 7 and 8 can, as noted above, also transmit data to two FIFOs. Refer to Figure 4; MAC 7 can transmit data to the computer workstation 2 via FIFO 11, to the VOIP telephone 1 via FIFO 13, or to both; MAC 8 can transmit data to the network 3 via FIFO 12, to the VOIP telephone 1 via FIFO 15, or to both.

Preferably, this flow of data is controlled by providing MAC 7 with a data bus that is connected to the input of both FIFO 11 and FIFO 13 and providing control signals on specific conductors for each of FIFOs 11 and 13 indicating whether the data on the data bus is intended for that FIFO, each of FIFOs 11 and 13 ignoring any data at its input when the control signals indicate that the data is for the other FIFO. Each of FIFOs 11 and 13 then only latches data that is intended for it. Alternatively, the FIFOs 11 and 13 may latch all data with the pointers in the FIFO being manipulated to purge unwanted data to prevent it from filling the FIFO buffers. It would be possible to allow data and control signals to be latched into FIFOs and to allow MAC 8 and buffer manager 9 to filter out the unwanted data, this is less advantageous than the other two arrangements as it risks filling the FIFOs with unwanted data. Broadcast packets received by MAC 7 are copied into both of FIFOs 11 and 13.

25

Any of the methods of controlling the flow of data from MAC 7 to FIFOs 11 and/or 13 discussed above can also be used for controlling the flow of data from MAC 8 to FIFOs 12 and 15. In the preferred embodiment of the invention, a single data bus is provided to pass data from MAC 7 to FIFOs 11 and 13 and from MAC 8 to FIFOs 12 and 15 with individual control signals indicating whether that data is intended for that FIFO.



The majority of the data flow will be between network 3 and computer workstation 2 and vice-versa. Consider data flowing from the computer workstation 2 to the network 3; a packet of data is transmitted to MAC 8 which routes that packet of data to FIFO 12 (and possibly also to FIFO 15). When that packet of data reaches the far end of FIFO 12, the data is passed to MAC 7. MAC 7 buffers that signal and drives the connection to the network. Similarly, a packet of data received at MAC 7 from the network 3 will be routed to FIFO 11 (and possibly also to FIFO 13). This packet of data will be passed to MAC 8 which buffers that signal and drives the connection to the computer workstation 2.

The situation described concerns the transfer of data between the computer workstation 2 and the network 3 when the VOIP telephone 1 is not in use. In those circumstances, the only delays introduced in the connection between the computer workstation 2 and the network 3 are caused by passing the data through MACs 7 and 8 and FIFOs 11 or 12. Thus the provision of VOIP telephone 1 does not significantly detract from the performance of the computer workstation when the VOIP telephone is not in use. Indeed, since the processing of data in the network extender module 6 will typically be much quicker than the transfer of data over the network 3, the delay in the transfer of data between the computer workstation 2 and the network caused by the introduction of the VOIP telephone 1 when that telephone is not in use can be made negligible. This is important because the bulk of the data transmitted will be between the computer workstation 2 and the network 3 and the user will not be aware of any significant degradation in performance as a result of the addition of the extender module and a VOIP telephone 1.

Since it is possible to receive more data for a port than it is able to transmit, flow control is provided in order to prevent packets from being lost due to the build-up of a backlog of packets of data awaiting transmission. In the preferred embodiment, the module includes means for sending a flow control packet to a data source to indicate that it should stop transmitting data to the module. This issue also has implications for the minimum size of the FIFOs 11 and 13. If the data source has started to send a packet of data to the module at the point at which flow control is deemed to be needed, the flow control packet must wait for that packet to be transferred before it can be sent. If that flow control packet arrives at the data source just after that source has started to send a packet, that packet will be sent before the data source is stalled. Thus two packets of data may need to be buffered after the point at which flow control is deemed necessary. Thus FIFOs 11 and 12 need to be relatively large. With data packets of 1.5 kilobytes (kB), FIFOs 11 and 12 should both be at least 6 kB.

20

An alternative embodiment of the invention, as shown in Figure 5, also has a MAC 7 for the network 3, a MAC 8 for the computer workstation 2 and two buffer managers 9 and 10 for the VOIP telephone 1. FIFOs 11 to 16 are, however, replaced with a single RAM 17. RAM 17 may be divided into a number of memory blocks, with each block representing one of the FIFOs 11 to 16. In this way, RAM 17 is used to implement the FIFOs of Figure 4 simply. Alternatively, the allocation of memory in the RAM 17 may be dynamic, memory representing FIFOs 11 to 16 being varied as required by the relative throughput of data. Such dynamic allocation of RAM could be implemented using a linked-list structure. For simplicity broadcast packets are copied to the lists for each (i.e. both of the two possible) destination.

The network extender module 6 shown in Figure 4 includes one FIFO for each data route shown in Figure 3. It may be advantageous to provide more FIFOs than this. For example, 5 some classes of data may be more important than others. In such circumstances each of the FIFOs in Figure 4 could be replaced with two FIFOs, one for priority data and one for less important data taking the same route. The FIFOs carrying important data would then be given priority over 10 other FIFOs. This is particularly advantageous for the VOIP telephone 1 as it would enable the VOIP data packets carrying voice information to be given priority over other VOIP data (such as control information bound for the network, for example, the telephone at the other end of a telephone call), 15 since it is particularly important for the voice information not to be delayed. Therefore it is advantageous (at the cost of extra circuitry) to provide two FIFOs for transferring packets from buffer manager 10 to MAC 7, the first of these for voice information and the second for other VOIP data 20 (e.g. control data) the first being given priority over the second.

Since the network extender module of the present invention may take the form of a separate module that can be added to a 25 pre-existing computer workstation, a convenient place for the network extender module is as part of a VOIP telephone unit. It could also be provided in the computer, for example as part of a network card in the computer. If the module is provided in the computer then the other conventional circuits 30 of the VOIP telephone could also be provided on the computer (e.g. on the network card).

The network extender module may be implemented in hardware. MACs 7 and 8 and buffer managers 9 and 10 may, for example,

be provided as ASIC modules, with the FIFO buffers being provided by the ASIC or by a separate memory block as shown in Figure 5.

TI-30405